REMARKS/ARGUMENTS

Amendments were made to the specification to correct errors and to clarify the specification. No new matter has been added by any of the amendments to the specification.

Claims 1-5, 7-10, 12-21 and 24 are pending in the present application. Claims 6, 11, 17 and 22-23 were canceled; Claims 1, 7-8, 12 and 18-19 were amended; and Claim 24 was added. Reconsideration of the Claims is respectfully requested.

Claims 1 and 12 have, inter alia, been amended to incorporate subject matter of now canceled Claims 6 and 17, respectively,

I. Rejection Under 35 U.S.C. § 101

In response to a rejection of Claim 23 under 35 U.S.C. § 101, Claim 23 has been canceled, and replaced by new Claim 24. Claim 24 positively recites a computer program product on a computer readable medium in a data processing system. Accordingly, the rejection under 35 U.S.C. § 101 has been overcome.

II. Rejection Under 35 U.S.C. § 112, Second Paragraph

In response to a rejection of Claims 11 and 22 under 35 U.S.C. § 112, second paragraph, Claims 11 and 22 have both been canceled. Accordingly, the rejection under 35 U.S.C. § 112, second paragraph, has been overcome.

III. Objection to Claims

Claims 7, 8 and 19 were respectively amended to overcome objections thereto.

IV. 35 U.S.C. § 102, Anticipation (*Havekost* Reference)

The Examiner rejected Claims 1-3, 5-14, and 16-23 under 35 U.S.C. § 102(b), as being anticipated by U.S. Patent Publication No. 2002/0108077, to *Havekost* et al. (hereinafter "*Havekost*"). This rejection is respectfully traversed.

V. 35 U.S.C. § 102, Anticipation (*Sauvage* Reference)

The Examiner rejected Claims 1-5, 8, 9, 12-16, 19, 20, and 23 under 35 U.S.C. § 102(e), as being anticipated by U.S. Patent Publication No. 2003/0056156, to *Sauvage* et al. (hereinafter "*Sauvage*"). This rejection is respectfully traversed.

VI. Teachings of Applicants

In making their invention, Applicants sought to enable devices such as mobile phones and personal digital assistants (PDAs) to be very simply managed and set up for use, while at the same time avoiding disruptions and requiring little or no effort by users. Applicants recognized that this objective could be achieved by managing a significant parameter associated with a resource or component in a device of the above type, wherein data associated with the parameter is held or contained in a data structure, and the parameter comprises a triplet of values. These parameter values are maximum and minimum values representing a range, and a variable. A second component is configured to interact with the data structure, to perform each of the tasks of accessing the data structure, monitoring the variable value, and responsive to the variable value lying within the range, managing the parameter, and selectively updating data in the data structure. These teachings of Applicants are set forth in the specification, such as at page 1, line 25 to page 2, line 18, and page 3, lines 1-7, which read as follows:

> Therefore, there is a need for a mechanism that allows devices to be simple to use, with little maintenance required by the user. There is also a need for a mechanism that allows a component (e.g. an application) to be managed without causing disruption to the normal running of the device.
>
> According to a first aspect, the present invention provides a system for managing at least one parameter associated with a first-component, wherein the at least one parameter comprises at least three values corresponding to a minimum value and a maximum value together representing a range and a variable value, the system comprising: a data structure comprising data associated with the at least one parameter, means for accessing the data structure, means for monitoring the variable value, and means, responsive to the variable value lying within the range, for managing the at least one parameter. **[Specification, page 1, line 25 to page 2, line 18.]**

> Aptly, there is provided means for updating the data structure with the data, when the first component is launched. More aptly, a second component comprises the means for accessing, the means for monitoring and the means for managing. Still more aptly, the system comprises means for notifying the second component of events associated with the first component. **[Specification, page 3, lines 1-7.]**

Claim 1 as now amended recites an embodiment of the invention as follows:

> 1. (Currently Amended) In a data processing system, a system for managing at least one parameter associated with a first component, wherein the one parameter comprises at least three values corresponding to a minimum value and a maximum value together representing a range and a variable value, the system comprising:
>
> a data structure holding data associated with the at least one parameter; and
>
> a second component separate from the data structure comprising, collectively, a means for accessing data of the data structure, means for monitoring the variable value,

and means, responsive to the variable value lying within the range, for managing the at least one parameter, and for selectively updating data in the data structure.

VII. Rejection of Claim 1 Based on *Havekost* Reference

In rejecting Claim 1 as being anticipated by *Havekost*, the Examiner stated the following:

> Referring to Claims 1, 12, and 23:
>
> a. In paragraph 0047, Havekost et al. disclose that modules may include respective logic that monitors the operational status of equipment controlled by each module, the values of process variables compared to a predetermined operating range (e.g., a setpoint or control range), or any other failure information desired (wherein the at least one parameter comprises at least three values corresponding to a minimum value and a maximum value together representing a range and a variable value).
>
> b. A data structure comprising data associated with the at least one parameter would be inherent to the system of Havekost et al. since the variable and operating ranges have to be stored in the control logic.
>
> c. In paragraph 0047, Havekost et al. disclose that each phase may, on a step by step basis, process failure information associated with the module or modules that carry out that step to determine if the step can proceed (means for accessing the data structure, means for monitoring the variable value, and means, responsive to the variable value lying within the range, for managing the at least one parameter).
>
> d. With respect to Claim 23, in paragraph 0058, Havekost et al. disclose implementing the invention in software. Referring to Claims 2 and 13, in paragraph 0047, Havekost et al. disclose monitoring the operational status of equipment (wherein the at least one parameter represents a resource associated with the system). **[Office Action dated September 27, 2006, pages 4-5.]**

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Anticipation focuses on whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. *Kalman v. Kimberly-Clark Corp.*, 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983). Moreover, it is a fundamental principle of patent law that prior art must be considered in its entirety. **MPEP 2141.02.**

Applicants respectfully submit that *Havekost* fails to teach every element of the claimed invention arranged as they are in Claim 1. For example, *Havekost* does not teach, in the overall combination of Claim 1, the following Claim 1 features:

1. "The data structure recited by Claim 1, in combination with the second component of Claim 1, accesses data of the data structure and selectively updates data therein"

2. The second component as recited by Claim 1.

VIII. Claim 1 Distinguishes over *Havekost*

The *Havekost* reference is generally directed to failure management for process control systems. Teachings of *Havekost* that are pertinent to Applicants' Claim 1 are found at paragraphs [0012], [0014], [0032], and [0047] through [0049], and at **Figure** 5. These sections of *Havekost* are as follows:

> **[0012]** A hierarchical failure management technique establishes run-time communication relationships between unit modules and control modules within a process control system so that failure information may be automatically passed from control modules to unit modules during execution of a process. More specifically, the hierarchical failure management technique described herein uses the run-time communication relationships between unit modules and control modules to enable a unit module to register interest in and to automatically receive and process failure information from control modules that are needed by the unit module to carry out a current phase or step of a process control algorithm and to ignore failure information associated with control modules that are not currently needed by the unit module to carry out a phase or step. (emphasis added)
>
> **[0014]** According to one aspect of the invention, a system and method for managing failures in a process control system collects failure information within each of a plurality of control modules and generates a composite failure code within each of the control modules based on the failure information collected by that control module. Additionally, the system and method automatically sends a group of composite failure codes to a unit module and determines whether a process should be stopped based on the group of composite failure codes.
>
> **[0032]** In operation, failure information may be generated by one or more of the field devices **40-50**, **60** and **61** in response to a control parameter falling outside of a desired or required operating range, in response to the failure of a device to perform an action, or in response to any other abnormal or unacceptable condition within the field device. For example, the field device **60** may be a level sensor that senses the level of a paint mixture within the mixing tank **62**. If the measured level of the paint mixture within the tank **62** exceeds the predetermined maximum level, the controller **36** may automatically receive failure information from the field device **60** indicating that the paint mixture in tank **62** has exceeded the predetermined maximum level. Also, for example, the smart field device **44** may be a valve that enables the colorant dispenser **66** to meter controlled amounts of one or more colorants into the mixing tank **62**. The field device **44** may include sensing facilities and control logic (e.g., confirm logic) that generates failure information in response to the failure of the colorant dispenser **66** to respond properly to commands and/or control signals sent by the field device **44**. For example, if the field device **44** sends a signal to the colorant dispenser **66** to open a valve within the dispenser **66** and the field device **44** detects (via wires, communications, etc.) that the valve has not opened or that the valve is stuck open, the field device **44** may generate appropriate failure information that is passed to the controller **36**.

[0047] In addition to carrying out actions invoked by phase steps or providing explicitly required status information, the modules **208-212** and **220** may include respective logic **222-226** that monitors the operational status of equipment controlled by each module, the values of process variables as compared to a predetermined operating range (e.g., a setpoint or control range), or any other failure information desired. Because the equipment hierarchy of the process control system **10** is fully integrated, the failure information collected by each of the modules **208-212** and **220** can, if desired, be automatically propagated up to the unit **200**. Thus, each phase may, on a step by step basis, process the failure information associated with the module or modules that carry out that step to determine if the step can proceed. In this manner, neither the phase nor the batch executive has to directly monitor and process all of the failure information on a system-wide basis, as was the case with prior systems. Instead, each phase can determine for each step whether the process can continue by processing only the failure information that pertains to that step at the time the step is executed (or immediately prior to the time that the step is executed). The batch executive may then receive, for example, a message from a phase that the process has been stopped, or may, alternatively, receive a message to take some other action if a failure has been detected but the process was not stopped as a result of the failure.

[0048] **FIG. 5** is an exemplary schematic block diagram that illustrates the manner in which control modules propagate failure information up to unit phases and the manner in which the unit phases may further process this failure information to determine whether or not the currently executing batch process should be stopped or whether some other action should be taken by the system operator. As shown in **FIG. 5**, a plurality of control modules **300-304** are associated with a phase running within a unit **306**. The steps of the phase may send parameters to one or more of the control modules **300-304** that are subservient to the unit **306** in a particular sequence at particular times to cause the modules **300-304** to carry out actions (e.g., to cause a piece of equipment to execute an action) that carry out the desired batch process. Alternatively or additionally, the phase may explicitly register interest in one or more of the control modules **300-304** that are not subservient to the unit **306** but which may affect the execution of one or more phase steps. In any event, each of the control modules **300-304** can automatically pass information such as, for example, requested failure information, to the unit phases. In particular, the control modules **300-304** include respective lists of failure information **308-312**. Each of the lists **308-312** includes failure information related to sensors, equipment, process control variables, etc. that are needed by a respective one of the control modules **300-304** to perform its function. For example, the lists of failure information **308-312** may include an indication that a sensor is no longer functioning, that a valve is stuck open or closed, that a control parameter has fallen outside of a predetermined range of values, or any other information that relates to the operational status of a piece of equipment and/or a control loop needed to carry out module functions. (emphasis added))

[0049] Importantly, the control modules **300-304** further include respective logic **314-318** that processes the failure information contained in the lists **308-312** to form a composite failure code for each of the control modules **300-304**. Generally speaking, the composite failure codes are formed and sent by each of the control modules **300-304** to the unit **306** as the unit phase enters or is about to enter a step that needs (i.e., implicitly requires the module to carry out actions or explicitly requires the module for safety purposes or any other purpose) that particular control module so that the unit phases can more quickly and easily process composite failure information associated with a relatively large number of devices, equipment, etc. (emphasis added)

Havekost, paragraphs [0012], [0014], [0032] and [0047] through [0049].

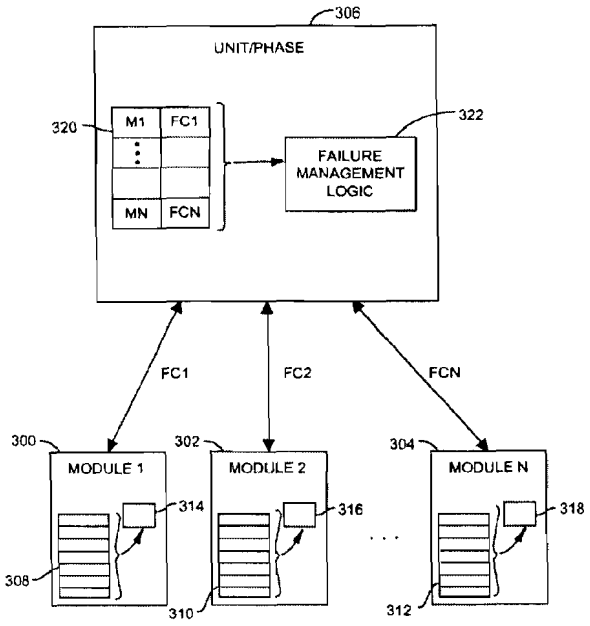

FIG. 5

Havekost, Figures 5

Paragraph [0012] of *Havekost*, which is the first paragraph of the summary thereof, emphasizes a central teaching of *Havekost*, that is, to pass failure information from a control module to a unit module. At paragraph [0014], *Havekost* teaches that a control module collects failure information, and then uses it to generate composite failure codes that are sent to a unit module. As taught at paragraph [0032], the failure information may be generated by field devices **40-50**, **60** and **61**, such as are used in connection with a paint mixing procedure. **Figure 5** shows control models **300-304** that are provided with logic **314-318** of the composite failure codes, as taught at paragraph [0049]. The failure codes are then sent to a unit or unit phase **306**, as further taught thereby. The unit phase processes the failure information, as taught by both paragraphs [0047] and [0048], in order "to determine whether or not the code currently executing that process should be stopped", or whether some other action should be taken.

It is readily apparent that *Havekost* fails to show a data structure, as taught by Claim 1, that interacts with a separate component capable of performing all the respective functions recited by Claim 1 for the second component. In the Office Action, it was stated that the data structure "would be inherent". However, the clear and fair teaching of *Havekost*, as taught by the sections thereof set forth above, is an arrangement wherein failure information originates with field devices **50-60** or the like, is routed to

control modules such as **308-312**, and the control modules send failure information, in the form of composite failure codes, to a unit module or unit phase such as unit module **306**. The unit module then makes decisions based on the failure codes. This arrangement is not considered to in any way disclose the data structure recited by Claim 1, wherein the second component thereof accesses data of the data structure, and selectively updates the data of the data structure. Moreover, the combination of the data structure and second component of Claim 1 would be of no benefit or relevance in the arrangement of *Havekost*.

Havekost likewise fails to show the second component as now recited by Claim 1. The Office Action indicated that logic included in modules, presumably meaning logic **314-318** of control modules **300-304**, could perform a monitoring function by monitoring operational status of equipment and various failure information. However, *Havekost* emphasizes, such as at paragraphs [0047] and [0048] as discussed above, that it is a unit module, such as unit phase **306**, that determines what action to take after receiving failure information from a control module. Thus, the clear teaching of *Havekost* is that the unit module, and not the control module, perform the management function. In contrast, Claim 1 teaches that monitoring and management functions are to be combined in a single second component. Moreover, *Havekost* does not show a control module interacting with a data structure that is separate from the second component, as is likewise recited by Claim 1.

IX. Claim 1 Distinguishes over *Sauvage*

As taught for example at paragraph [0009] and [0054], *Sauvage* is directed to an arrangement for monitoring a number of different time dependent variables of a CPU or the like, by procuring a range of expected values of a variable having upper and lower limits. Figure 1 shows the arrangement of *Sauvage* including, inter alia, value monitors **13**, **14** and **15**, a value processor **16**, an alarm generator **20** and a comparator element **34**.

Claim 1 as now amended is considered to distinguish over *Sauvage*, particularly in reciting the second component thereof in the overall combination of Claim 1. As stated above, Claim 1 has been amended to incorporate the subject matter of original Claim 6, now canceled. In the Office Action, the Examiner did not cite *Sauvage* against Claim 6. Accordingly, Claim 1 as amended clearly distinguishes over *Sauvage*. *Sauvage* fails to show or suggest a single component that is separate from a data structure, and collectively comprises means for accessing data of the data structure, means for monitoring the variable value, and means, responsive to the variable value lying within the range, for managing the at least one parameter.

X. Remaining Claims Distinguished over the Cited References

Independent Claims 12 and 24 respectively incorporate subject matter similar to the patentable subject matter of Claim 1, and are considered to distinguish over the art for at least the same reasons given in support thereof.

Claims 2-5 and 7-10, and Claims 13-16 and 18-21, depend from Claims 1 and 12, respectively, and are each considered to distinguish over the art for at least the same reasons given in support thereof.

Claims 4 and 15 are additionally considered to distinguish over the art in reciting the feature that the action comprises a re-launch of the first component. In the Office Action, *Havekost* was not cited against these claims, and therefore does not show this feature.

XI. Conclusion

It is respectfully urged that the subject application is patentable over *Havekost* and *Sauvage* and is now in condition for allowance.

The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: December 27, 2006

Respectfully submitted,

/James O. Skarsten/

James O. Skarsten
Reg. No. 28,346
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Attorney for Applicants